

Logging using SLF4J

Exercise 1: Logging Error Messages and Warning Levels

Task: Write a Java application that demonstrates logging error messages and warning levels using SLF4J.

```
1 package com.example;
2
3 import org.slf4j.Logger;
4 import org.slf4j.LoggerFactory;
5
6 public class LoggingExample {
7     private static final Logger logger = LoggerFactory.getLogger(LoggingExample.class);
8
9     public static void main(String[] args) {
10         logger.error("This is an error message");
11         logger.warn("This is a warning message");
12     }
13 }
```

```
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA\bin\idea-agent-21.jar" -ea -Djbr.conf=C:\Program Files\JetBrains\IntelliJ IDEA\conf -jar C:\Program Files\JetBrains\IntelliJ IDEA\bin\idea-21.jar
17:38:50.611 [main] ERROR com.example.LoggingExample - This is an error message
17:38:50.615 [main] WARN com.example.LoggingExample - This is a warning message
```

Exercise 2: Parameterized Logging

Task: Write a Java application that demonstrates parameterized logging using SLF4J.

```
1 package com.example;
2
3 import org.slf4j.Logger;
4 import org.slf4j.LoggerFactory;
5
6 public class ParameterizedLoggingExample {
7     private static final Logger logger = LoggerFactory.getLogger(ParameterizedLoggingExample.class);
8
9     public static void main(String[] args) {
10         String userName = "Sai";
11         int age = 25;
12
13         logger.info("User {} has logged in.", userName);
14         logger.debug("User {} is {} years old.", userName, age);
15     }
16 }
```

```
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.2.5\lib\idea_rt.jar=1741:06.461 [main] INFO com.example.ParameterizedLoggingExample - User Sai has logged in.
17:41:06.470 [main] DEBUG com.example.ParameterizedLoggingExample - User Sai is 25 years old.
```

Exercise 3: Using Different Appenders

Task: Write a Java application that demonstrates using different appenders with SLF4J.

```
1 <configuration>
2   <appender name="console" class="ch.qos.logback.core.ConsoleAppender">
3     <encoder>
4       <pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{36} - %msg%n</pattern>
5     </encoder>
6   </appender>
7
8   <appender name="file" class="ch.qos.logback.core.FileAppender">
9     <file>app.log</file>
10    <encoder>
11      <pattern>%d{yyyy-MM-dd HH:mm:ss} %-5level %logger{36} - %msg%n</pattern>
12    </encoder>
13  </appender>
14
15  <root level="debug">
16    <appender-ref ref="console" />
17    <appender-ref ref="file" />
18  </root>
19 </configuration>
```

```
1 package com.example;
2
3 import org.slf4j.Logger;
4 import org.slf4j.LoggerFactory;
5
6 public class AppenderLoggingExample {
7     3 usages
8     private static final Logger logger = LoggerFactory.getLogger(AppenderLoggingExample.class);
9
10    public static void main(String[] args) {
11        logger.info("This message should appear in both the console and app.log");
12        logger.warn("This is a warning written to both appenders");
13        logger.error("This is an error message with appenders");
14    }
15 }
```

```
"C:\Program Files\Java\jdk-21\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2023.2.5\lib\idea_rt.jar=1743:03.016 [main] INFO com.example.AppenderLoggingExample - This message should appear in both the console and app.log
17:43:03.019 [main] WARN com.example.AppenderLoggingExample - This is a warning written to both appenders
17:43:03.019 [main] ERROR com.example.AppenderLoggingExample - This is an error message with appenders
```